David Fanning wrote:
> Nick writes:
>
>> I have regular arrayed data (1440*720). I'd like to change this data
>> to 360*180 array.
>> So I use 'Rebin' function.
>> But these data have NaN value.
>> If I use Rebin and there is a NaN value, new array becomes also NaN.
>>
>> For example, if there is only one NaN in the old array, the new-array
>> becomes NaN. But I want to make a new array except NaN data. This
>> situation makes residual data wasteful.
>>
>>   A = [1.5,2.5,3.6,4,7,8.8,9.0,!values.f_nan]
>> print, rebin(A, 1)
>> ;result is 'NAN'
>> ;That I expected value is mean(A, /nan)
>>
>> Is there any know-how to change array except NaN?
>
> Well, you seem to know how to change your array.
> Why don't you just find the NANs, change them to
> what you want them to be, then do the REBIN?

I haven't had to do this with IDL, so I didn't realize that IDL
handled it inconveniently. What I would normally want to have happen
when re-binning data with NaNs is that every element of the output
array whose calculation involved one of the NaN's in the input array
would itself contain a NaN, while all the other elements of the array
would be calculated normally. That's not something that could be
achieved by the approach you suggest. If that's a feature not already
provided as an option by rebin, it should be.